TECHNICAL REPORT
NATICK/TR-08/006

AD_______________

SOLDIER FLEXIBLE PERSONAL DIGITAL ASSISTANT PROGRAM

by
Mark Price
Jason Woytowich
and
Marc Carlson

InHand Electronics, Inc.
Rockville, MD 20850

January 2008

Final Report
December 2006 – September 2007

Approved for public release; distribution is unlimited

Prepared for
U.S. Army Natick Soldier Research, Development and Engineering Center
Natick, Massachusetts 01760-5020

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* 08-01-2008	**2. REPORT TYPE** Final	**3. DATES COVERED** *(From - To)* 22 Dec 2006 - 30 Sept 2007

4. TITLE AND SUBTITLE
SOLDIER FLEXIBLE PERSONAL DIGITAL ASSISTANT PROGRAM

5a. CONTRACT NUMBER
W911QY-07-C-0022

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Mark Price, Jason Woytowich and Marc Carlson

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
InHand Electronics, Inc.
30 W. Gude Drive
Suite 550
Rockville, MD 20850

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Natick Soldier Research, Development and Engineering Center (NSRDEC)
Kansas St. , ATTN: AMSRD-NSC-TS-S (Dennis Magnifico)
Natick, MA 01760-5020

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
NATICK/TR-08/006

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The main goal of the Soldier Flexible Personal Digital Assistant Program was to develop prototypes of a novel flexible display technology device for demonstration in a laboratory setting and use in Future Force Warrior (FFW) demonstrations. This device was designed to meet the needs of FFW and to provide situational awareness, mapping software, and other data for the soldier. The key unique feature of the Soldier Flexible PDA Program is the integration of state-of-the-art flexible display technology. This technology, available in different forms from E-Ink and from the U.S. Army Flexible Display Center at Arizona State University, can be used to create novel packages and mechanical enclosures that better conform to body-worn computing requirements and also have relatively low power requirements. The reduction of the size and weight of batteries needed will lengthen mission profiles and also improve body-worn requirements.

15. SUBJECT TERMS
MAPPING PROTOTYPES DATA DISPLAYS INTEGRATED SYSTEMS INFORMATION RETRIEVAL
TEXTILES ELECTRONICS DISPLAY SYSTEMS WEARABLE COMPUTING SITUATIONAL AWARENESS
BATTERIES LIGHTWEIGHT SOFTWARE TOOLS FFW(FUTURE FORCE WARRIOR)
EMBEDDING DIGITAL MAPS FLEXIBLE DISPLAYS PDA(PERSONAL DIGITAL ASSISTANT)

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	SAR	34	Dennis Magnifico, Program Manager
U	U	U			**19b. TELEPONE NUMBER** *(Include area code)* (508) 233-6944

Table of Contents

Table of Contents (Cont'd)

List of Figures

List of Tables

Preface

The Soldier Flexible Personal Digital Assistant (PDA) development effort, also known as the "Mainstrike" Program during development at InHand Electronics, consisted of a joint development between several parties. Research was conducted during the period December 2006 – September 2007 under contract number W911QY-07-C-0022. Mainstrike is the InHand codename used to designate the NSRDEC PDA designed to work with the Flexible Display Center's (FDC) FDC03A and FDC07 displays and this terminology can be used interchangeably throughout the report. This fieldable prototype was designed to meet the requirements set forth in Appendix A, as well as the contractual specifications arrived at during the course of the development. Programmatically, the execution of the final deliverable has involved multiple parties, whose roles are summarized below:

NSRDEC – Program coordinator and sponsor
InHand Electronics, Inc. (IHE) – Prime contractor for the development effort
E-Ink Corporation– subcontractor to InHand who supplied the development kits for the FDC display technology and the Actel controller chips to drive the display.
Artisent, Inc. – subcontractor to InHand who supplied the industrial design and consulting advice for FFW connectors and interfacing.
FDC – subcontractor to Natick and partner to InHand. FDC supplied the display panels and customized the flex circuitry to meet InHand's packaging requirements.
CERDEC – supplied the C2MINCs software.

A table providing the key project milestones is shown below:

Task	Completion Date	Notes
Kick-off Meeting	1/12/07	InHand internal kick-off
Block Diagram	2/2/07	System diagram, based on decisions from kick-off meeting and initial research
Requirements Documentation	2/16/07	A list of requirements for the prototypes agreed to be all parties
Specifications Documentation	3/2/07	A high-level system specification document describing design approach for most risky design components
Preliminary Design Review (PDR)	3/26/07	Review of design approach and initial documentation
Comprehensive Design Review (CDR)	4/26/07	Review of detailed design approach and documentation
Breadboard Prototype	6/21/07	Open-frame demonstration at InHand of flex display working with electronics
Functional Prototypes (4)	6/29/07	Delivery of packaged prototypes (3 shipped to NSRDEC; 1 delivery-in-place)
Functional Prototypes (2)	6/29/07	Delivery of packaged prototypes (deliver in place)
Final Documentation	9/30/07	Relevant prototype documentation

Soldier Flexible Personal Digital Assistant Program

1. Overview

The Soldier Flex PDA consists of a combination of technologies: a custom-designed external shell, designed in concert with Artisent, a Commercial-Off-the-Shelf (COTS) mainboard from InHand know as the Fingertip4, a custom-designed daughter card to support the Mainstrike functionality, and external cabling to support the FFW infrastructure.

A block diagram of the system is shown in Figure 1.

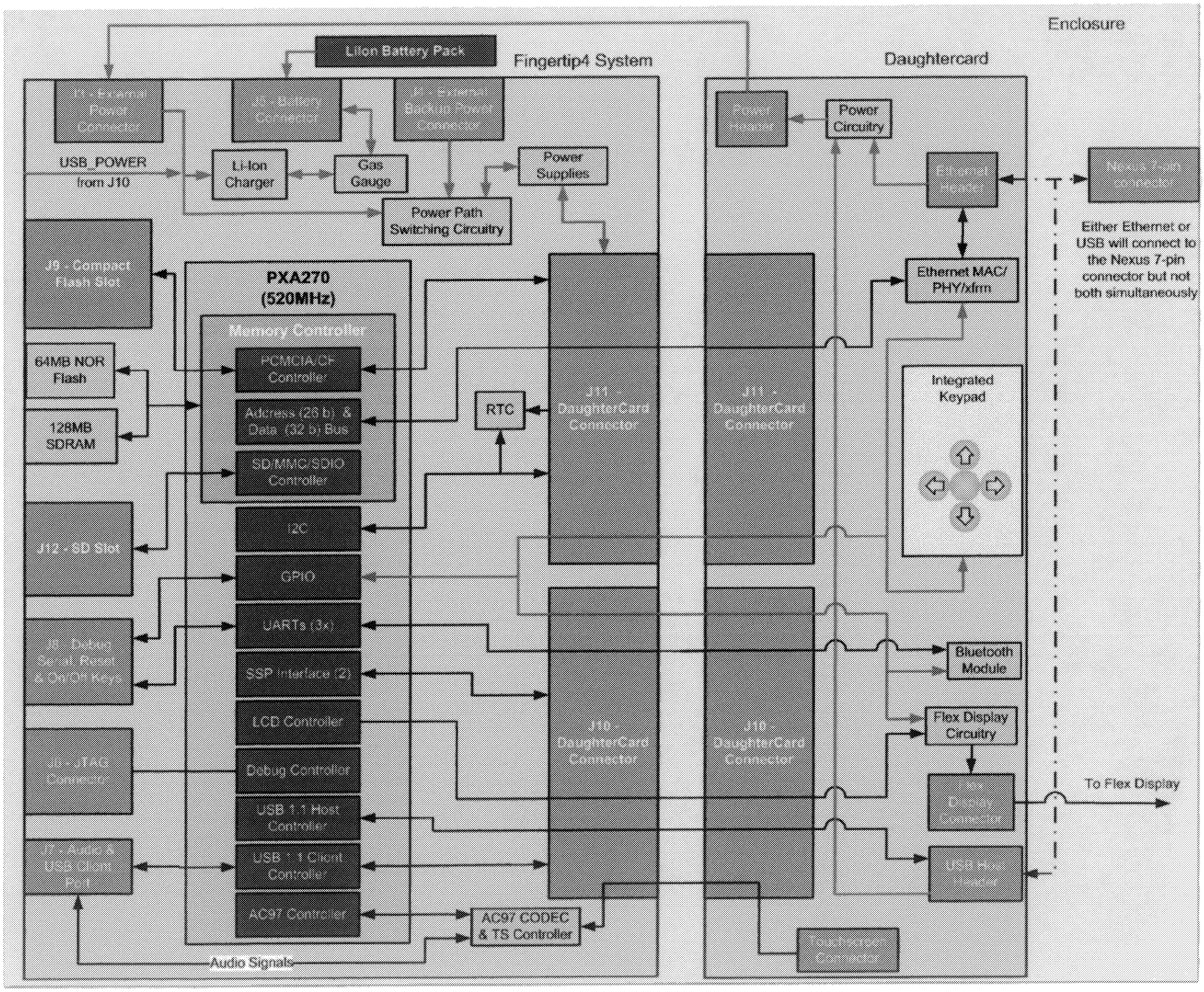

Figure 1. Mainstrike System Diagram

The custom daughter card interfaces to the Fingertip4 and provides the following services:

- Ethernet
- USB Host
- Touchscreen Connector
- Keypad
- Bluetooth
- Display

The components enabling these functions were inserted into a custom enclosure designed to meet the additional needs of the FFW Users. The mechanical design effort to achieve this requirement is detailed in the following section.

2. Mechanical Design

A significant portion of the Mainstrike effort dealt with the mechanicals associated with the design. This included coming up with requirements analysis, user comments, concept sketches, and finally digital rendering. The final mechanical design is highlighted later in this section, but Figure 2 shows the shape selected after PDR and final discussion.

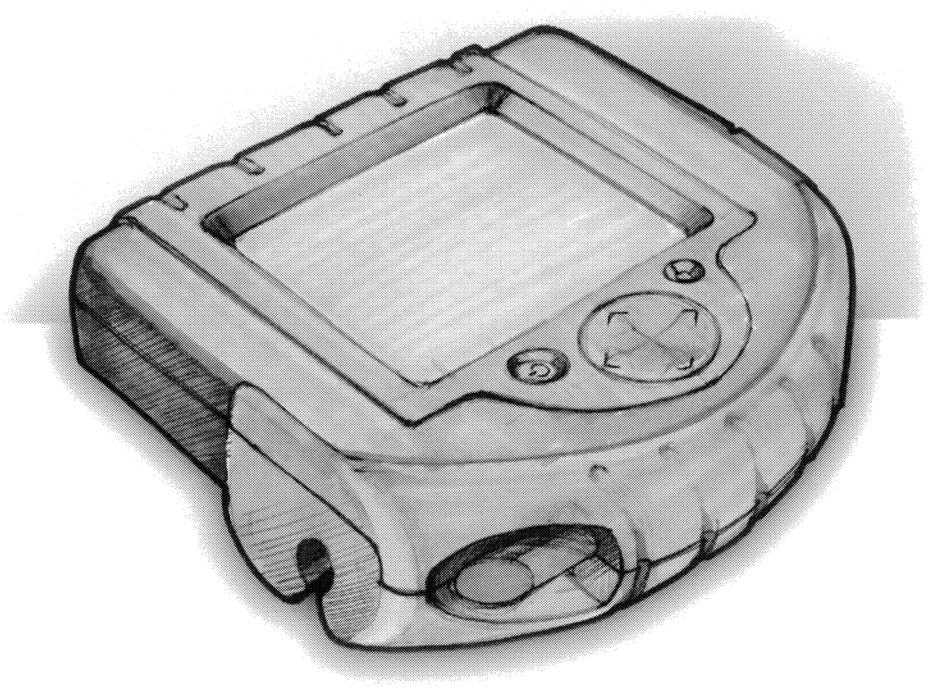

Figure 2. Sketch of Selected Mainstrike Design Concept

InHand worked with Artisent on the initial industrial design, and then used design files to create a package that could be manufactured using low-volume techniques (silicone molds). Based on feedback from the initial molding, we slightly altered the overmold to provide better tactile feel for the fixed buttons on the unit.

We also acquired styli, lanyards, mounting hardware and other miscellaneous system components (such as development cabling and test pouches) necessary to assemble the complete system and ready the units for field use.

Due to imperfections in the SS process (all stainless steel displays exhibited some degree of line-outs and other issues) for the displays, NSRDEC decided to use the silicon displays in two of the three delivered units.

All delivered units were configured with the custom Gunze touchscreens (flexible) and the moisture barrier process on the flex displays.

The mechanical design consisted of several custom pieces, the chief ones being:

- Flex display (stainless steel substrate)
- Flex touchscreen (polycarbonate substrate)
- External hard shell (with overmolding and integrated buttons)
- External cabling
- Internal Electronics (Mainboard, daughter card, interconnections, and buttons)

Figure 3 shows the modification of the FDC03A panel, necessary to package the display in a tightly-confined space using the Fingertip4. This panel, referenced as the FDC07 panel from E-Ink/FDC, was made using both Silicon (initially) and Stainless Steel (final delivery).

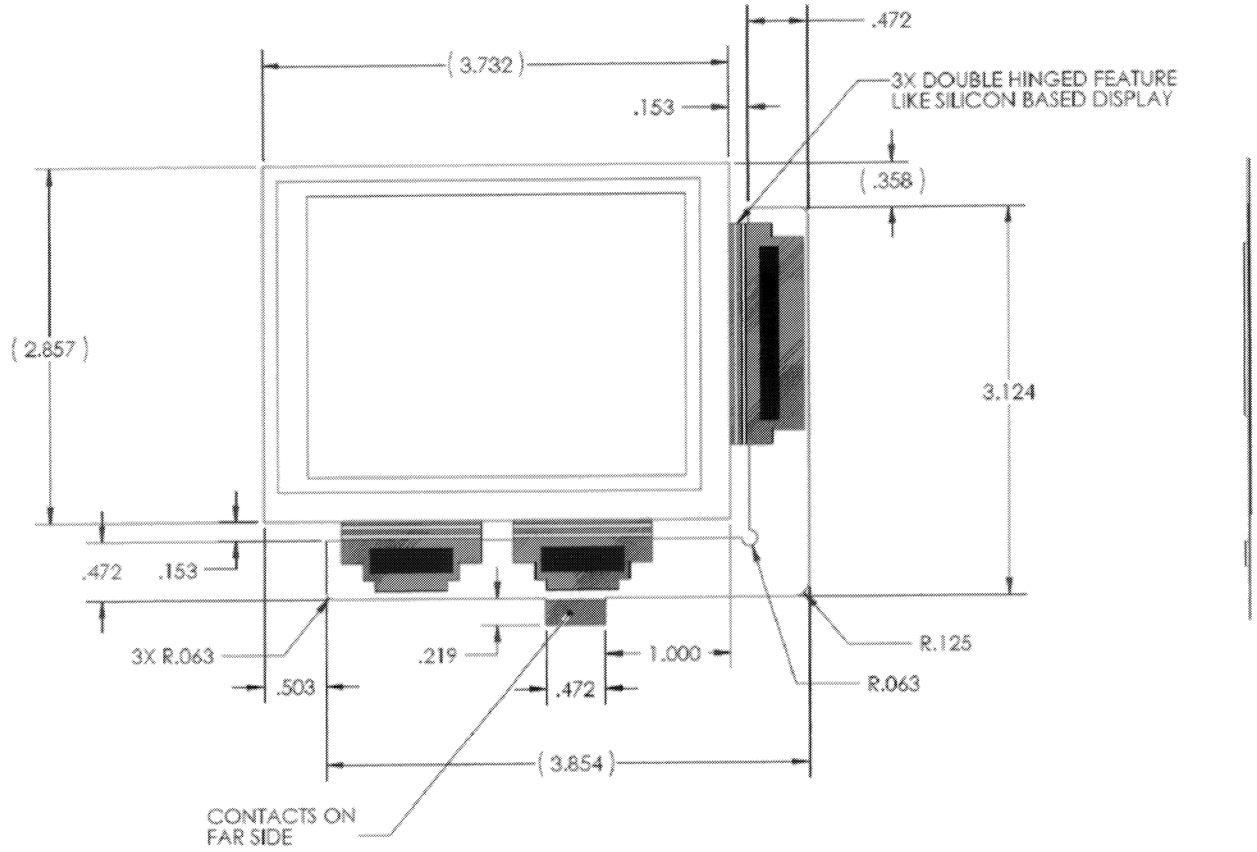

Figure 3. FDC03A with InHand-specified Flex

In order to design the external shell, first the internal stackup for the flex display and Fingertip4 needed to be designed. Figure 4 and Figure 5 give the dimensions and layout for these pieces.

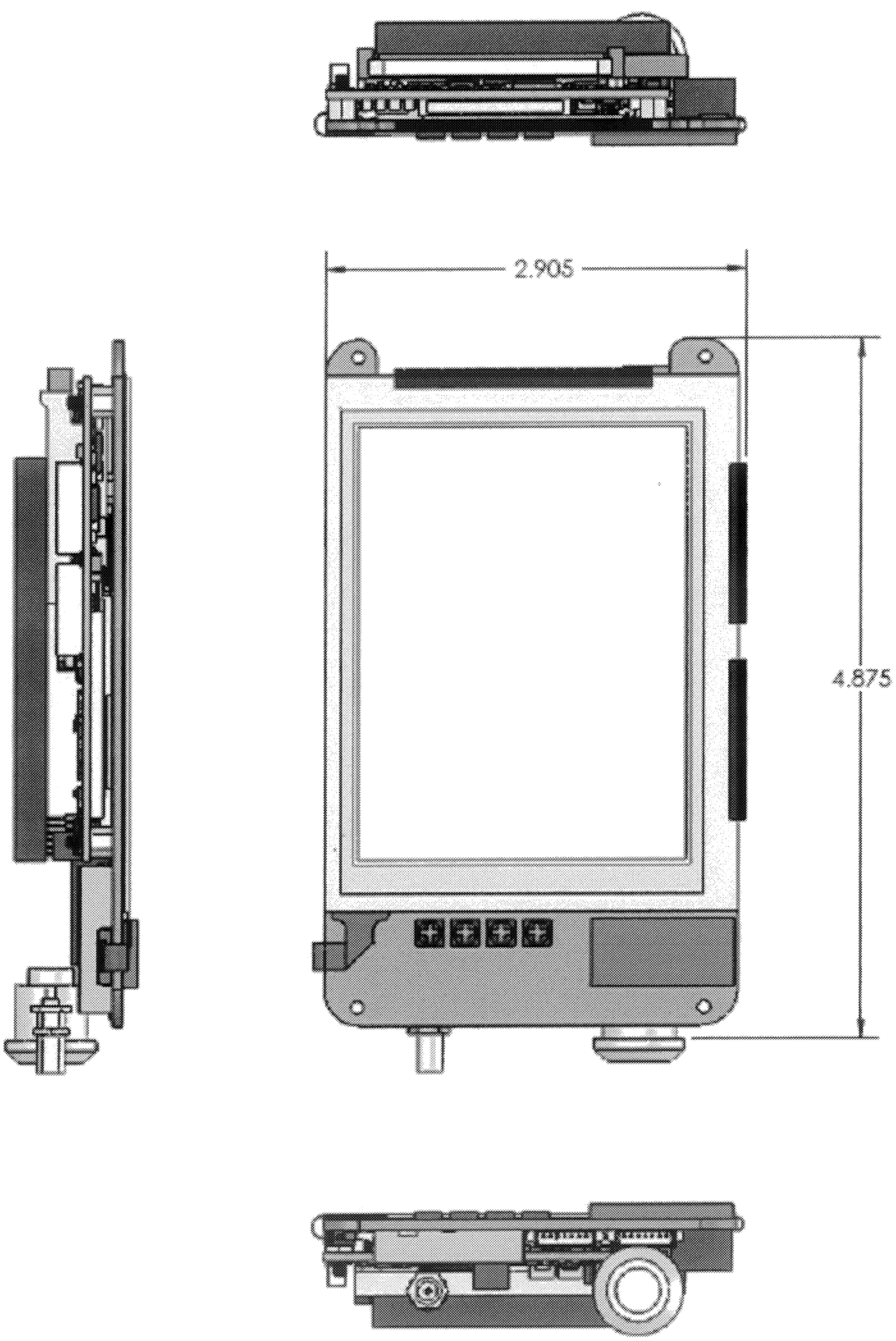

Figure 4. eDrawing of Mainstrike Internal Stack-up

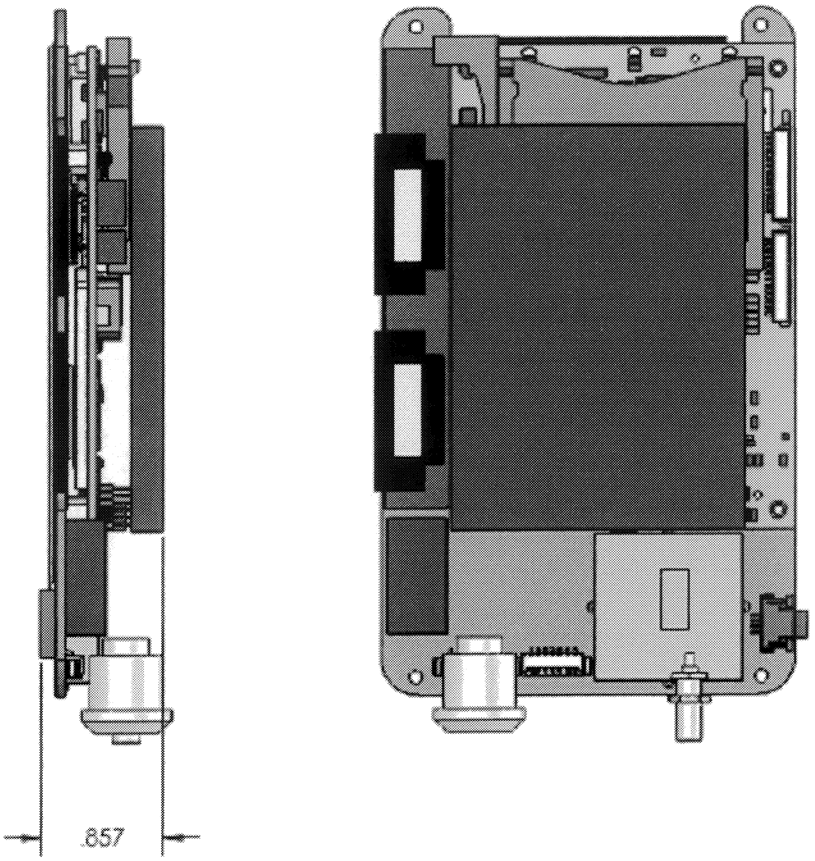

Figure 5. Mainstrike Internals

Once the initial internal stackup was complete, this allowed iterating on the external mechanicals, as well as finalizing the flex display stackup with the touchscreen. The mechanical variables involved in determining the flex stack up and final thickness can be seen in Figure 6.

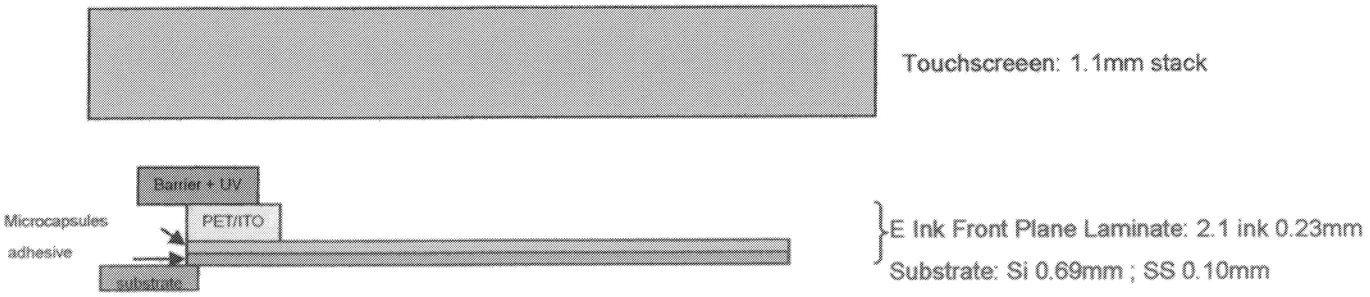
Touchscreeen: 1.1mm stack
Barrier + UV
Microcapsules
PET/ITO
adhesive
substrate
E Ink Front Plane Laminate: 2.1 ink 0.23mm
Substrate: Si 0.69mm ; SS 0.10mm

Figure 6. Flex Display Thickness Stackup

With the finalization of the internals and flex display stackup, this enabled InHand and Artisent to create the external casing design and specify the layout for the custom daughter card electronics. Figure 7 shows the rendering for the final external design.

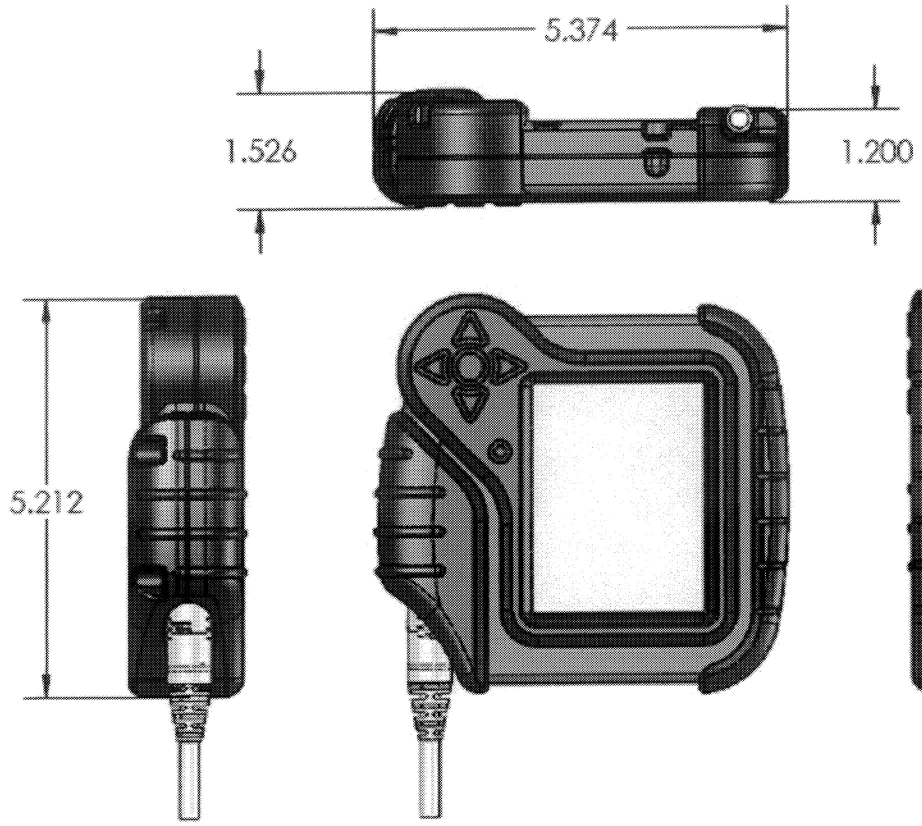

Figure 7. Final Mainstrike Design

Figure 8 shows the outline for the Mainstrike Printed Circuit Board (PCB) assembly. This outline represents the maximum area as defined by the mechanical packaging.

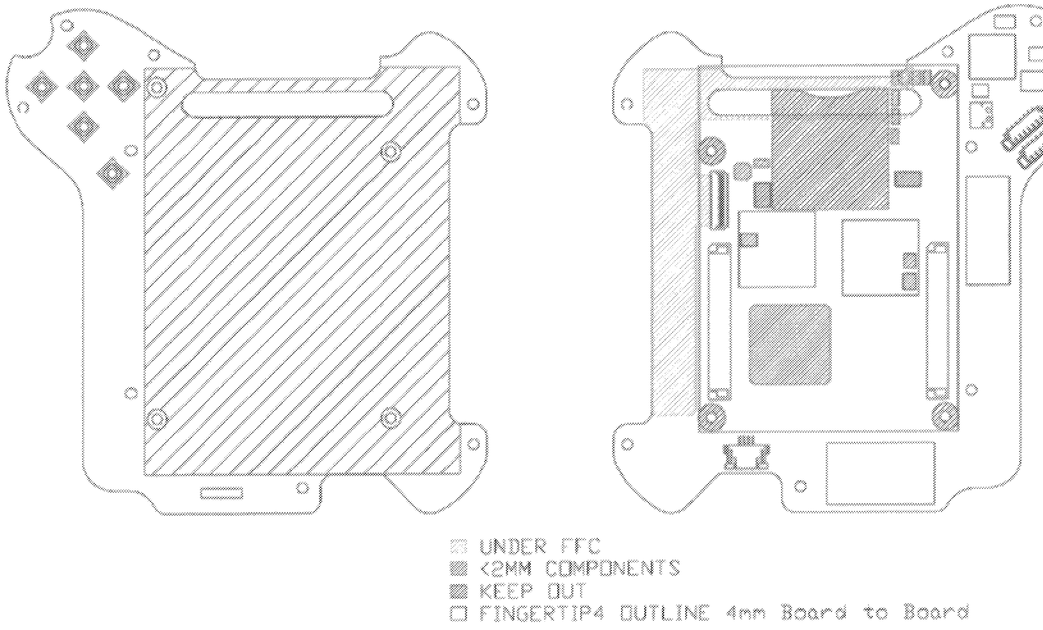

Figure 8. Preliminary Mainstrike Mainboard PCB Layout

These elements define the Mainstrike mechanicals. The subsequent sections (as well as the Appendices) highlight the design detail for the electronics and software.

3. Daughter Card Design

The electronic hardware performed an orderly migration from design through development.

Specific work included incorporating a Bluetooth module from A7 Engineering. The search for Class 3 Bluetooth modules produced no workable results, but as the A7 module could be attenuated in software, there was no need to perform hardware accommodations for this capability.

For the Flex display itself, we received and reviewed schematics from the FDC03A module and evaluated the proposed FDC07 InHand flex-cable.

The daughter card development process included component selection and finalizing schematic capture prior to CDR. We also fully updated our Mentor library with new components that were identified during final schematic modifications. We completed parts placement and layout of the Mainstrike PCB and fully kitted all components to build the required deliverables. Further, we successfully fabricated the PCB, electrically tested the board, and populated the 1st-article for engineering evaluation and test.

Unfortunately, we discovered two fabrication errors and one design error. The most serious fabrication error was in the antenna section of the A7 Bluetooth module. The layout in this area is critical for proper RF transmission and operation of the module, and required the use of blind vias. Although the Gerbers are correct, these blind vias were not fabricated correctly, so there is no contact with the ground plane. This did not affect our ability to bring up the rest of the board (and may not even affect the operation of the attenuated class 3 module), but was not as designed. The other errors were minor, but we decided to do a quick-turn of a new PCB to ensure a more robust product when fielded.

In order to highlight the advantages of the flex display technology, we concluded that a plastic touchscreen was necessary to demonstrate the durability of the display. As there are no standard touchscreens that exist compliant with the size and other specifications of the FDC flex display, we had a custom touchscreen fabricated by Gunze.

Per our CDR schedule that was worked out between InHand and FDC/E-Ink, we received an electrically functioning flex display that was used in the open-frame demo. Although delayed from early forecasts, we did receive a fully-functioning flex display on a Stainless Steel substrate.

The following sections highlight the Mainstrike architecture and the details of the functional modules.

3.1 Power Architecture

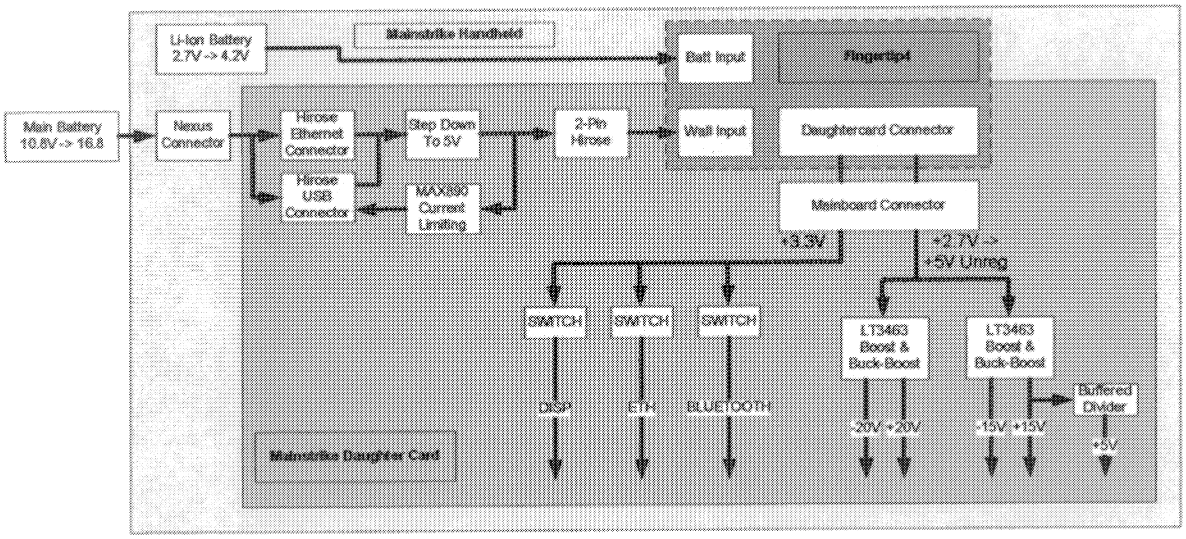

Figure 9. Power Architecture

The Mainstrike unit was designed to integrate into the FFW infrastructure, so the cabling, power, and software requirements were designed to meet this need. Although power is provided externally from the Nexus connector when connected to the FFW infrastructure, the Mainstrike unit was additionally designed to allow independent operation through an internal Li-Ion battery. When provided externally, the power is cabled to the daughter card from the Nexus connector through either the USB connector or the Ethernet connector. The origin (through the FFW battery) is a BB-2950 battery which has a voltage output from 10.8V to 16.8V with 16V being the nominal output. Alternate configurations of this battery are able to output up to 33.8V and the system is able to accommodate this even though it is not expected in operation of the handheld. The external power is stepped down to 5V on the daughter card and then cabled from a 2-Pin Hirose to the Fingertip4's wall power input. The pinout is shown in Table 1.

Table 1. Power Output Connector Pinout

Pin	Name	Function
1	GND	Ground
2	VWALL	+5V

The main board connector is bypassed to prevent unnecessary constriction of the current. The Fingertip4 is then responsible for recharging the Li-Ion battery and providing power to the daughter card. The 5V daughter card power is also used to provide power over the USB host port. There is no path from the Li-Ion battery to the 5V power so it should be noted that without external power being added to the system, the USB host port will not have the needed 5V to power slave devices. This is not a problem because the USB function is provided via the Nexus connector.

The battery running this device when not connected via the Nexus connector is a single cell 1800mAh Li-Ion battery. This is connected directly to the Fingertip4 so that the onboard power supply will power the system through the daughter card connectors. It is an Ultralife S00062.

The Fingertip4 generates a regulated +3.3V from any available source and supplies it to the daughter card over the main board connectors along with the unregulated power from either the Li-Ion battery or the +5V stepped down. The +3.3V, used to power much of the board, is capable of being switched on/off for the Ethernet controller, Bluetooth module and display controller. The unregulated voltage is used to power the high gate and source drive voltages (±20 and ±15.)

3.2 Main Board Connectors

The daughter card is connected to the Fingertip4 main board through two Hirose FX8-100P-SV1(91) connectors. These differ from the connectors used on a standard daughter card in that they increase the total stacking height by 1mm. The higher connector is being used to allow more components to fit between the Fingertip4 and the daughter card. The pinout for these is available in the Fingertip4 user manual.

3.3 Ethernet

Ethernet is being provided by an SMSC LAN9118 connected directly to the system bus on chip select 2. It is a 10/100baseT MAC+PHY selected because we have already proved the design in Linux. Several GPIOs are used to assist in configuring the controller. These are listed in Table 2.

Table 2. Ethernet GPIOs

Signal	Function
nETH_PWR_EN	Enable/disable power to the Ethernet subsystem
nETH_RESET	Reset the Ethernet controller
ETH_INT	Ethernet controller interrupts

The recommended magnetic component for this controller is the Bel Fuse S558-5999-46. A Hirose DF13-7P-1.25V(20) is used as the Ethernet connector on the daughter card. The pinout is shown in Table 3.

Table 3. Ethernet Connector Pinout

Pin	Name	Function
1	TX+	Controller TX Positive
2	TX-	Controller TX Negative
3	RX+	Controller RX Positive
4	NC	No Connect
5	RX-	Controller TX Negative
6	BATT-	Battery return
7	BATT+	Battery power

3.4 USB Host

The USB host capabilities are provided directly by the PXA270. The +5V power rail provides the power and is current limited to 700mA by a MAX890.

The same type of connector as the Ethernet controller is used for USB Host. The pinout is shown in Table 4.

Table 4. USB Host Connector Pinout

Pin	Name	Function
1	D+	USB Data Positive
2	D-	USB Data Negative
3	NC	No Connect
4	Gnd	USB Ground
5	Pwr	USB Power (+5V)
6	Batt-	Battery return
7	Batt+	Battery power

3.5 Bluetooth

Bluetooth support is provided by the A7 eb100-HCI module. This has a UART based HCI using BCSP for which there is good support for in the Linux BlueZ Bluetooth stack. This is connected to the PXA270's Bluetooth UART.

It is a class 2 Bluetooth device with an effective range of 10m or more. The intended application is in the range of a class 3, low power, Bluetooth device. The output power is decreased programmatically over the BCSP to bring the module to the power range of a class 3 device.

One signal is used to activate/deactivate the Bluetooth module and it is listed in Table 5.

Table 5. Bluetooth GPIOs

Signal	Function
nBT_PWR_EN	Enable/disable power to the Bluetooth subsystem

3.6 Display Controller

The 8Track Display controller connects directly to the PXA270's LCD controller. Commands and data are sent to the controller through the LCD controller's data lines usually reserved for raw image data. The controller's 1.5V core voltage is provided by an LDO from the switch +3.3V source for the display. Source and Gate driver voltages are provided by two LT3463 Boost/Buck-Boost switching regulators fed from the unregulated supply. An additional +5V supply is added by simply buffering the output of a voltage divider off of the +15V rail. The controller takes the pixel, line and frame clocks and outputs data and clocks appropriate to drive the source and gate drivers.

The VCOM voltage is supplied from a buffered voltage divider adjusted with a SMT potentiometer. This can be varied from -5V to +5V and once set and packaged, it is not able to be adjusted.

The display is connected to the daughter card through an OMRON XF2B-3945-31A. This provides all power and signals needed to drive the FDC03A or FDC07 displays. The pinout is provided in Table 6.

Table 6. Display Connector Pinout

Pin	Name	Function
1	V_SOURCE-	Negative voltage for the source drivers (-15V)
2	V_SOURCE+	Positive voltage for the source drivers (+15V)
3	+5V	+5V Gate keeper voltage, not needed for the MX860 source drivers
4	VDD	Logic voltage (+3.3)
5	GND	Ground
6	NC	Formerly source driver left/right select
7	SDLE	Source driver latch enable
8	SDDO0	Source driver data
9	SDDO1	
10	SDDO2	
11	SDDO3	
12	SDDO4	
13	SDDO5	
14	SDDO6	
15	SDDO7	
16	SDCLK	Source driver clock
17	SDCE0	Source driver chip enable
18	SDOE0	Source driver output enable
19	SDCE1	Source driver chip enable
20	SDOE1	Source driver output enable
21	SDCE2	Source driver chip enable
22	SDOE2	Source driver output enable
23	SDCE3	Source driver chip enable
24	SDOE3	Source driver output enable
25	SDCE4	Source driver chip enable
26	SDOE4	Source driver output enable
27	VCOM	Display common
28	VCS	Display common
29	GND	Ground
30	NC	Formerly the border drive voltage
31	V_GATE+	Gate driver positive voltage (+20V)
32	V_GATE-	Gate driver negative voltage (-20V)
33	NC	Formerly gate driver logic voltage
34	GND	Ground
35	GDOE	Gate driver output enable
36	NC	Formerly gate driver right/left select
37	GDSP	Gate driver start pulse
38	GDCLK	Gate driver clock
39	GND	Ground

3.7 Keypad

The pushbuttons for the keypad are located on the daughter card. The keys implemented are: Up, Down, Left, Right, Select and Power and are accessible as GPIOs. There is also a reset button that is not part of the keypad.

3.8 Touchscreen

The PDA makes use of a custom Gunze 3.8" 4-wire resistive, plastic touchscreen. The connector is a Molex 52207-0485.

4. Software Design

There were two main areas of software development: porting the C2MINCs software to the Mainstrike platform and writing the device drivers and embedded code for the Mainstrike daughter card.

Due to the toolchain and versioning differences between the Recon C2MINCS software implementation and the InHand implementation, InHand elected to acquire a Recon unit to assure an understanding of the package used with CERDEC. Further, InHand opened up a dialogue with CERDEC personnel to gain insight on reconfiguring the C2MINCs software to better map to the Mainstrike device.

Due to the delay in starting the main effort, InHand front-loaded the C2MINCs port using a standard InHand platform, but once the team was fully engaged through adding the necessary subcontractors, further progress on the C2MINCs port was suspended until we got the Mainstrike hardware operational, as per the original project plan.

We also received an alternate application (PDF viewer and manuals) from CERDEC that, although not a contracted activity, provided an excellent demonstration of the Flex display technology.

In terms of driver development, the key drivers were the development of the flex display API and the Bluetooth Module. On the driver side, we validated the A7 Bluetooth module under Linux using the FT4.

The most challenging work, however, was the development of the Linux frame buffer.

The C2MINCs software continued to be an area of frustration in making scheduled progress. For C2MINCs, the GUI requires Qtopia, which is not part of the standard FT4 Linux distribution. This required an additional porting effort and the acquisition of the necessary Qtopia licenses. Although we ported a preliminary version of C2MINCs, continued versioning problems existing in the Qtopia port and the C2MINCs libraries curtailed a successful bring-up on the FT4. We worked with CERDEC on receiving updated libraries, but without a direct relationship with CERDEC to push this through, we were unsuccessful in getting timely and appropriate support.

As part of our contingency plan for software development, we also received and successfully ported an alternate application (PDF viewer and manuals) from CERDEC.

By project end, we had all the device drivers fully tested and ported, as well as an alternate application for demonstrating the flex display technology. We were unsuccessful in completing the C2MINCs port, but as this was never a contracted activity beyond moving over the binaries, we instead provided an SDK to allow CERDEC to perform the port themselves.

4.1 Operating System

The Fingertip4 is running a Linux 2.6 kernel. Drivers and board startup code have been customized to interface specifically with this board.

4.2 Ethernet

The Linux drivers for the SMSC LAN9118 are included as a module. Only a minor fix had to be made to the distributed code to allow for compilation. The board startup code is responsible for powering on and resetting the controller by controlling GPIOs.

4.3 USB Host

The USB Host capabilities have been included as modules in the system. Only drivers for mass storage devices have been included.

4.4 Bluetooth

The A7 Bluetooth module runs on a UART interface available in the standard BlueZ Bluetooth protocol stack. A startup script is responsible for powering on the module and attaching to it using the BlueZ utilities.

4.5 Display

A frame buffer driver which interfaces with the E-Ink 8Track controller through the PXA270 LCD controller has been written. The purpose of this driver is to send the appropriate commands and data to update the display only when the frame buffer has been written into. The user is given a piece of memory, a frame buffer, which writes data as if it were an 8bit, palletized image. The memory is controlled so that any access, which will normally be writes for a frame buffer, notifies the driver through the linux page fault mechanism. After accesses stop, the image is copied into a second memory location which includes commands and data.

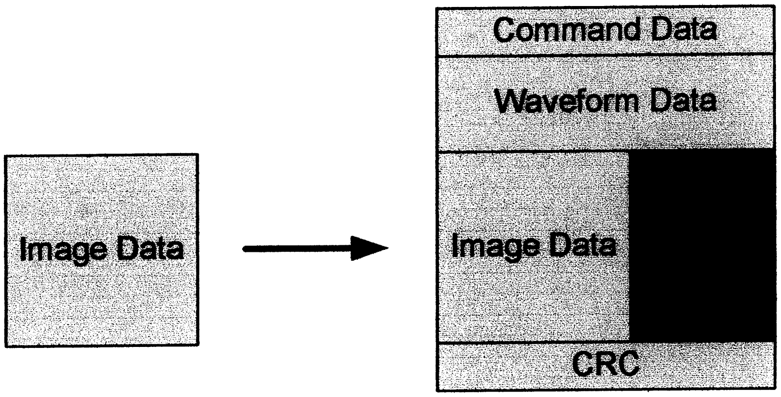

Figure 10. 8Track Frame Structure

The following commands, listed in Table 7, must be supported.

Table 7. 8Track, Needed Commands

Name	Function
Powerup	Give powerup timings for the displays Gate and Source driver voltages
Config	Configure display size, padding, etc.
Init	Set operating modes
Display	Display a new image

Each of these commands can be fixed for the device and the CRC can be disabled.

Each image update consists of a number of frames sent with the same display command. The voltage that is applied to each pixel on the display is defined by the waveform data which is nothing more than a lookup table indexed by the frame number (within the current image update) and the 8-bit pixel value. The waveform is updated to match the palette chosen by the application.

The data is output to the 8-Track display controller in 16-bit chunks but the width of the data must be wider than half the actual image width to allow all source drivers to completely fill with data.

4.6 Keypad

A small keypad driver which maps GPIOs to arbitrary input codes as a Linux generic input device has been written. This map is hard coded into the board startup code and cannot be fixed. The mapping is shown in Table 8. The linux key code conversions can be found in include/linux/input.h in the kernel source.

Table 8. Key Scan Code Conversion

Key	GPIO	Linux Name	Linux Code
Up	22	KEY_UP	0x067
Down	23	KEY_DOWN	0x06C
Left	24	KEY_LEFT	0x069
Right	25	KEY_RIGHT	0x06A
Select	26	BTN_SELECT	0x13A
Power	0	KEY_POWER	0x074

4.7 Qtopia

Qtopia has been provided as the user's primary interface with the system. The package was built from Qtopia PDA Edition 2.2.0. There were modifications to the build scripts to allow it to be built with the Arm Embedded Linux Development Kit (ELDK.) Multi threading and run-time type information (rtti) are non-standard features that have been enabled. Tslib was used to provide touch screen support and a custom Qtopia module was written to connect to the keypad input driver described in 4.6. A change to the command Qtopia uses to go to sleep was needed, as it assumed that APM was enabled and it is not.

4.8 Application Software

The C2MINCs software is provided as binaries and has yet to be run on the system

There is a PDF viewer that runs with no problems on the system. It can be launched from the Qtopia desktop.

Appendix A: Technical Specification

1. Mainstrike System Specifications

1.1 CPU

Marvell PXA270, 520 MHz internal clock speed.

1.2 Flash Memory

64MB MB Intel P30 StrataFlash® memory

1.3 SDRAM

128MB of Mobile SDRAM with self-refresh mode.

1.4 Displays

FD07 320 x 240 (QVGA) electrophoretic flex display

1.5 Peripherals

- One slot Secure Digital (SD) Memory Card connector (internal)
- 5-way navigation keypad
- USB host
- 10/100 Base T Ethernet
- High precision on-board Real Time Clock (RTC).
- Low power, class 3 Bluetooth transceiver, ~1m range

1.6 Operating Systems

Linux (releases 2.6)

1.7 Battery

1800 mAhr, thin profile, single cell Li-Ion battery pack

1.8 Battery Charger

On board, single-cell Li-Ion battery charger providing up to 900mA charge current

1.9 External Power Supply and Battery

16VDC external power input

1.10 System Power Consumption (including display)

- 60 mW typical in sleep.
- 600 mW, running without Ethernet
- 1 W running with Ethernet

1.11 Standby & Run Time

- Over 4 days standby
- 6.5 hours run time.

1.12 Mechanical Dimensions

5.32" (w) x 5.17" (l) x 1.54" (d) (135.2mm x 131.5mm x 39.2mm)

1.13 Weight

13 oz (0.37 kg)

This document reports research undertaken at the U.S. Army Natick Soldier Research, Development and Engineering Center, Natick, MA, and has been assigned No. NATICK/TR-08/006 in a series of reports approved for publication.

Appendix B: User's Manual

1. Overview

Mainstrike is a PDA designed to work with the displays produced at the Flexible Display Center at Arizona State University and integrate with the FFW system. It was designed to meet the requirements set forth in MAI-IPM-100 (Appendix A).

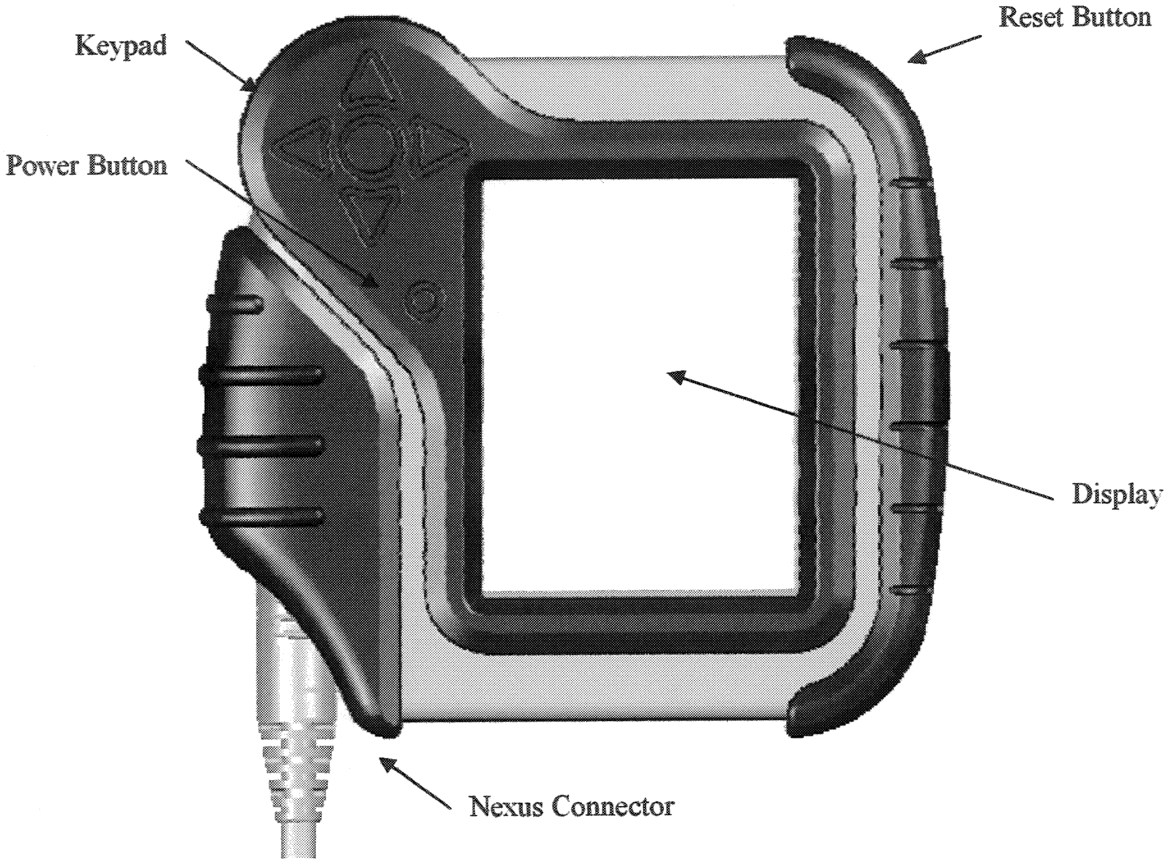

Figure 11. Mainstrike PDA

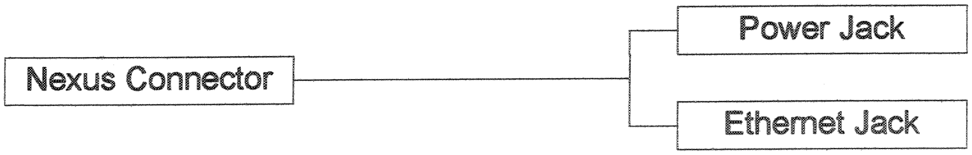

Figure 12. Ethernet Development Cable - Block Diagram

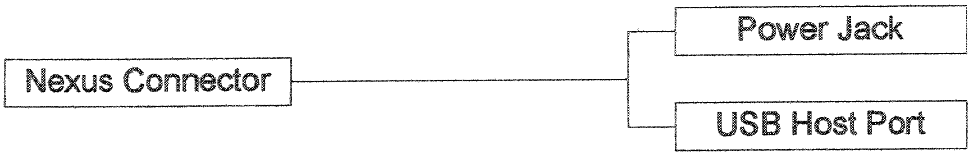

Figure 13. USB Development Cable - Block Diagram

2. Power

The PDA is powered either internally from a Li-Ion battery or externally through the Nexus connector. It is expected that when external power is applied it is coming from either the FFW system, supplying 16V nominal, or from one of the Ethernet/USB cables provided for development. Although it is possible to charge a USB enabled PDA using one of the Ethernet cables it is not recommended as connecting the PDA to the incorrect upstream device could possible damage both devices. Both development cables need to have the 9V wall-wart power supply connected to the Power Jack to operate/recharge the battery. The associated service does not need to be connected for the power to get to the device.

3. Operating System

The PDA is running a Linux 2.6 kernel. The kernel, all libraries and applications have been developed using the Embedded Linux Development Kit (ELDK) version 4.1 with arm-linux-x86. Note that this is different than the arln-linux-x86-ulibc version and applications compiled using one will not work with the either.

4. Desktop Environment

Qtopia is used as the desktop environment. Note that Qtopia has not been designed to work with low frame rate displays and will refresh the screen every time the minute changes on the clock or any other graphics need updating.

5. Storage

Inside of each PDA is an SD Card which provides the system with 2GB of storage. This is recommended for the storage of the majority of large documents. The root storage system is mostly completely filled by the operating system (OS) and Qtopia.

6. User Interfaces

6.1.1 Touchscreen

There is a touch screen which allows the user to interact with the Qtopia environment. A stylus is provided on the opposite side of the device, near the reset button. Currently the Qtopia calibration utility is ineffective at re-calibrating the device and it is recommended that if this is necessary to save the previous settings located in /etc/pointercal.

6.1.2 Keypad

There are 5 keys provided to assist the user in navigating through Qtopia and other applications. In the upper-left hand corner of the device are the up, down, left, right and select buttons. These allow you to sort through menu options or select applications and documents from the desktop without using the touch screen

6.1.3 Power and Reset Buttons

The reset button in the upper-right hand corner of the device is recessed and accessed using the stylus. This should be used only during development or if the device truly locks up.

The power button wakes/sleeps the device from Qtopia. Currently there is no indicator that the device is asleep except for unresponsiveness from the keypad. When woken up, the display will flash several times before showing the desktop.

7. USB

On a USB enabled PDA it is possible to transfer files to/from the PDA using any USB Mass Storage (UMS) device such as a USB flash drive. To activate the USB port the USB development cable must be plugged into the Nexus connector and the 9V wall-wart power supply must be connected. The USB device will not be powered unless there is external power available.

When the UMS device is plugged into the unit an icon should appear in the lower left hand corner indicating that it has been recognized. All of the files on the drive should be visible in the Documents pane of the Qtopia desktop. When all desired transfers have completed, tap on the USB icon and select the option to "Unmount" the UMS device. Once this is done it is safe to remove the device. If the icon does not disappear immediately, perform the procedure again without re-inserting the drive.

8. Ethernet

Ethernet is the primary communication mechanism for the PDA. An Ethernet enabled PDA connects to either the FFW system through the Nexus connector or to a development system through a hub.

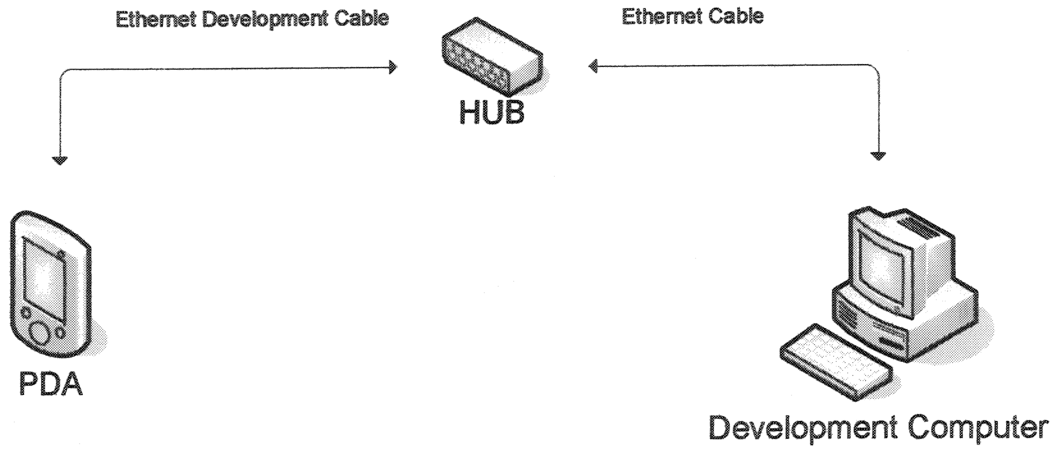

Figure 14. Development System Connection

It is possible to connect to the PDA in a development environment without the use of a HUB. If the Ethernet port on the development computer supports Auto cross-over, the Ethernet development cable can be plugged directly into the PC. If not a manual cross-over can be introduced by bridging the two with a crossover cable.

The IP address of the PDA can be set to either a default IP or to auto-configure using DHCP. To statically set the IP of a PDA create a text file in /etc called static_ip which contains nothing but the desired address. Ex. "192.168.1.100" This is the address acquired after reset. The default configuration is to have the IP statically configured to 192.168.1.100. During operation, the IP can be changed using the onscreen terminal and the ifconfig command. Ex. "ifconfig eth0 192.168.0.101"

Communication with the PDA should be done using ssh (Secure Shell) and scp (Secure Copy.) Both commands require you to supply a username and password. The defaults for these are shown below:

Username: root
Password: inhand

Please refer to the man pages for both of these commands on a standard Unix/Linux system to become familiar with the commands.

Although it is not required, it is highly recommended to have power available to the system when communicating over Ethernet. When connected to the FFW system it is impossible to communicate without the power available, but in a development environment the 9V wall-wart should be utilized.

9. Bluetooth

Bluetooth is automatically enabled and configured on hci0 during power on. All of the standard BlueZ commands and libraries for configuring/controlling bluetooth can be used. The startup script is located in /etc/init.d/bluetooth.

10. Display

The display is a bi-stable electrophoretic, flexible display developed the FDC in conjunction with E-Ink. The display has an extremely low refresh rate and the current implementation of the display drivers must completely flash black and white before settling on the desired image. It is possible for there to be "ghosting" or remnants of a previous image faintly visible in the background. It is possible to fix this using the eighttrack_init command which will tell the display to flash black and white several times in succession in order to erase the image. On the next screen update the image should be clearer.

List of Acronyms

CDR	Comprehensive Design Review
CERDEC	US Army Communications-Electronics Research, Development, and Engineering Center
COTS	Commercial-off-the-Shelf
FDC	Flexible Display Center
FT4	Fingertip4
FFW	Future Force Warrior
IHE	InHand Electronics, Inc.
NSRDEC	US Army Natick Soldier Research, Development, and Engineering Center
PDA	Personal Digital Assistant
PDR	Preliminary Design Review
USB	Universal Serial Bus

Distribution List

One report to each:
Dennis Magnifico, Program Manager
U.S. Army Natick Soldier RDEC
ATTN: AMSRD-NSC-TP-S
1 Kansas Street
Natick, MA 01760-5056

Henry Girolamo
U.S. Army Natick Soldier RDEC
ATTN: AMSRD-NSC-TP-S
1 Kansas Street
Natick, MA 01760-5056

Darlene Rideout
U.S. Army Natick Soldier RDEC
Natick Contracting Division
1 Kansas Street
Natick, MA 01760-5056

Dr. David Morton
Army Research Laboratory
M/S AMSRL-SE-EO
2800 Powder Mill Rd
Adelphi, MD 20783-1145

David Mark
U.S. Army Natick Soldier RDEC
Natick Contracting Division
1 Kansas Street
Natick, MA 01760-5056

Dr. Eric Forsythe
Army Research Laboratory
M/S AMSRL-SE-EO
2800 Powder Mill Rd
Adelphi, MD 20783-1145

Roger Masadi
U.S. Army Natick Soldier RDEC
Future Force Warrior
1 Kansas Street
Natick, MA 01760-5056

Electronic Distribution follows:

Henry.Girolamo@us.army.mil
Darlene.Rideout@us.army.mil
Dennis.Magnifico@us.army.mil
David.morton1@us.army.mil
Eric.forsythe@us.army.mil
Roger.masadi@us.army.mil